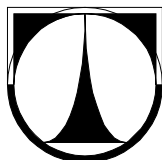


TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky a mezioborových studií



DIPLOMOVÁ PRÁCE

Liberec 2010

Ondřej Valenta

TECHNICKÁ UNIVERZITA V LIBERCI
Fakulty mechatroniky, informatiky a mezioborových studií

Studijní program: N2612 – Elektrotechnika a informatika
Studijní obor: Informační technologie

**Testování distribuovaného rozpoznávacího systému
v reálných podmínkách**

**The Testing of Distributed Recognition System in Real
Conditions**

Diplomová práce

Autor: Bc. Ondřej Valenta
Školitel: Ing. Miroslav Holada, Ph.D.
Konzultant: Ing. Josef Chaloupka, Ph.D.

Originál zadání

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé diplomové práce a prohlašuji, že s o u h l a s í m s případným užitím mé diplomové práce (prodej, zapůjčení apod.).

Jsem si vědom toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

V Liberci dne: 6.9. 2010

Podpis:

Poděkování

Chtěl bych vyjádřit své poděkování vedoucímu mé diplomové práce, Ing. Miroslavu Holadovi, Ph.D. za odborné vedení, cenné rady a přínosné konzultace.

Abstrakt

Cílem této práce je navrhnout klientské testovací aplikace, které budou využívat distribuovaný rozpoznávací systém izolovaných slov vyvinutý na Technické univerzitě v Liberci. Tento rozpoznávač pracuje na principu klient – server, kde komunikace mezi oběma stranami probíhá po internetu. Tyto aplikace, ovládané pouze hlasem, by měly umožnit testování tohoto systému v reálných podmínkách vybranou skupinou uživatelů. Jejich názor bude získán pomocí dotazníku.

Klíčová slova:

distribuovaný rozpoznávač

klientská aplikace

rozpoznávání řeči

dotazník

Abstract

The aim of this thesis is to develop the testing applications, which will be using distributed recognition system for isolated words. This system was developed on Technical university in Liberec. This recognition system works on client - server basis, where the communication between both sides proceeds via internet. These voice controlled applications should allow test of this system in real condition by selected group of users. Their opinion will be picked up by the questionnaire.

Keywords:

distributed recognition system

client application

speech recognition

questionnaire

Obsah

PROHLÁŠENÍ	3
PODĚKOVÁNÍ	4
ABSTRAKT	5
ABSTRACT	6
OBSAH	7
1 DISTRIBUOVANÝ ROZPOZNÁVACÍ SYSTÉM	9
1.1 FUNKCE	9
1.2 REÁLNÝ PROVOZ	12
1.2.1 Vliv internetového připojení.....	12
1.2.2 Vliv použitého vybavení.....	12
1.2.3 Vliv klientské aplikace	13
1.2.4 Vliv okolního prostředí.....	13
1.2.5 Vliv způsobu promluvy uživatele	13
2 KLIENTSKÉ APLIKACE	14
2.1 OBECNÁ PRAVIDLA	14
2.1 VYUŽITÍ APLIKACÍ	16
2.1.1 Desktopové aplikace.....	16
2.1.2 Mobilní aplikace	17
2.2 VELIKOST A STRUKTURA SLOVNÍKU.....	17
2.3 ŘÍZENÍ DIALOGU	18
2.4 DETEKCE A KOREKCE CHYB	19
2.5 ZPĚTNÁ VAZBA.....	21
3 NÁZOR UŽIVATELE.....	22
3.1 HODNOCENÍ ODBORNÍKY	23
3.2 HODNOCENÍ UŽIVATELI	23
3.3 PROTOTYP.....	24
3.3.1 Wizard of Oz.....	24
3.4 DOTAZNÍK.....	25
3.4.1 Výběr a skladba otázek.....	25
3.4.2 Příklad dotazníku.....	26
OBRÁZEK 5 PŘÍKLAD DOTAZNÍKU UVEDENÉM V [14]	27
3.5 VÝBĚR HODNOTITELŮ	27
3.6 PŘÍKLADY EXISTUJÍCÍCH METOD	27
3.6.1 SUMI.....	27
TABULKA 3: PŘÍKLADY TVRZENÍ V DOTAZNÍKU SUMI	28
3.6.2 SUXES.....	28
OBRÁZEK 6: HODNOTÍCÍ PROCEDURA METODY SUXES [13]	29
4 TESTOVACÍ APLIKACE	30
4.1 KONCEPCE PROGRAMU	30
OBRÁZEK 7: STRUKTURA TESTOVACÍ APLIKACE	30
4.1.1 VCB soubor	31
TABULKA 4: SEZNAM ZNAČEK A JEJICH VÝZNAM	31
4.1.2 Režimy aplikace	32
4.2 GRAFICKÉ ROZHRANÍ.....	32
OBRÁZEK 8: OKNO APLIKACE	33
5 TESTY	34
5.1 SADA TESTŮ	34
OBRÁZEK 9: POHYB OBJEKTEM PO OBRAZOVCE VE ČTYŘECH SMĚRECH	35
OBRÁZEK 10: PROHLÍZEČ FOTOGRAFIÍ	35
OBRÁZEK 11: UKÁZKA TESTŮ ZOBRAZENÍ NÁZVŮ HLAVNÍCH MĚST A ZMĚNA BARVY OBRÁZKU ...	36
OBRÁZEK 12: VYPLNĚNÍ DOTAZNÍKU	36
5.2 VÝBĚR UŽIVATELŮ	37
5.3 DOTAZNÍK.....	37
OBRÁZEK 13: DOTAZNÍK	39
6 VÝSLEDKY	40

6.1 ČTENÍ TEXTU	40
6.2 ZMĚNA BARVY OBRÁZKU	40
6.3 POHYB OBJEKTEM PO OBRAZOVCE	40
6.4 ZOBRAZENÍ NÁZVŮ HLAVNÍCH MĚST	40
6.5 PROHLÍŽENÍ OBRÁZKŮ	41
6.6 VYPLNĚNÍ FORMULÁŘE	41
6.1 REAKCE PROGRAMU	42
OBRÁZEK 14: NÁZOR UŽIVATELŮ: REAKCE PROGRAMU JSOU OKAMŽITÉ	42
6.2 PROGRAM REAGUJE ZCELA NEČEKANĚ	42
OBRÁZEK 15: NÁZOR UŽIVATELŮ: PROGRAM REAGUJE ZCELA NEČEKANĚ	42
6.3 OVLÁDÁNÍ VYŽADUJE VELKÉ SOUSTŘEDĚNÍ.....	43
OBRÁZEK 16: NÁZOR UŽIVATELŮ: OVLÁDÁNÍ VYŽADUJE VELKÉ SOUSTŘEDĚNÍ.....	43
6.4 NEVÍM, JAK VYSLOVOVAT DANÉ FRÁZE	43
OBRÁZEK 17: NÁZOR UŽIVATELŮ: NEVÍM, JAK VYSLOVOVAT DANÉ FRÁZE.....	43
ZÁVĚR.....	44
LITERATURA.....	45
PŘÍLOHY V ELEKTRONICKÉ FORMĚ.....	47
DVD-R	47

Úvod

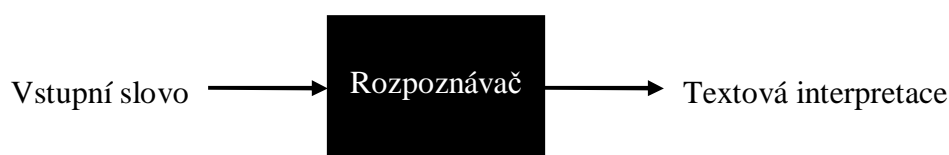
Aplikace využívající ovládání hlasem se čím dál častěji dostávají k uživatelům ať už v podobě informačních služeb, ve výuce nebo prostřednictvím různých her. Stále častěji pronikají i do přenosných zařízení, jako jsou mobilní telefony. Rozpoznávače řeči se stále zdokonalují a v praxi se s nimi setkávají specialisté z oborů jako je např. lékařství či soudnictví, kde pomáhají odbourat náročné zapisování údajů. Samozřejmostí je využití lidmi s tělesným postižením, pro které je jiný způsob ovládání počítače obtížný nebo nemožný.

Zatím však nebyl vyvinut rozpoznávač pracující s nulovou rozpoznávací chybou. Stále se tedy setkáváme se situacemi, kdy jako například chybně rozpoznány slovy, která znějí podobně, zachycení slova, aniž by bylo nějaké vysloveno nebo naopak vyslovený povel nevyvolá požadovanou akci. Chování rozpoznávače ovlivňují faktory, jako je hladina okolního hluku, vady řeči uživatele, nesprávně nastavené parametry záznamu hlasu, které při dnes zažitém způsobu ovládání aplikací pomocí standardních vstupních zařízení (klávesnice, myš) nehrají roli. Co se od standardního vstupu dále liší je rozdílná obtížnost pro rozpoznávač z hlediska jazyka, pro který je rozpoznávač trénován. Například anglický jazyk je z pohledu obtížnosti rozpoznávání řeči méně náročný než jazyk český, který na rozdíl od uváděného anglického jazyka obsahuje velké množství tvarů jednoho slova (z tohoto pohledu je problémem časování, skloňování apod.) Chyby způsobené rozpoznávačem proto musí být ošetřeny vhodně navrženým rozhraním, dostatečnou zpětnou vazbou, skladbou slovníku a další řadou prvků tak, aby se možnost chybné funkce aplikace snížila na minimum.

Práce se zabývá testováním rozpoznávače izolovaných slov, kdy je vybraným uživatelům nejprve představen specifický způsob ovládání dané klientské aplikace, poté jsou jim zadány jednoduché úkoly, které by měli uživatelé provést. Nakonec je uživatel dotázán na jeho názor ohledně způsobu práce s danou testovací aplikací. Jako forma sběru dat byl zvolen jednoduchý dotazník v papírové formě čítající kolem deseti položek, ve kterém uživatel zaškrtně jemu nejbližší názor. Výsledný koncept aplikace byl navržen a jeho komponenty realizovány na softwarové platformě .NET. K vývoji bylo použito prostředí Microsoft Visual Studio 2008 a jako programovací jazyk byl zvolen Visual C++.

1 Distribuovaný rozpoznávací systém

Pro splnění zadání této práce je využito již hotového jádra klientské aplikace, která komunikuje po internetu se vzdáleným serverem, na kterém běží samotný rozpoznávač. Princip činnosti (jak funkce řečového rozpoznávače, tak i samotné komunikace mezi klientem a serverem) není předmětem této práce. K celému procesu rozpoznávání je možné přistupovat jako k „černé skřínce“, kde na vstupu je akustický signál (slovo nebo fráze) a na výstupu textová interpretace tohoto slova, jak je znázorněno na obrázku 1.



Obrázek 1: Pohled na úlohu rozpoznávání

Rozpoznávač izolovaných slov pro Český jazyk byl vyvinut na Technické univerzitě v Liberci v roce 2001 laboratoří počítačového zpracování řeči SpeechLab [1]. Byl použit v mnoha aplikacích, např. v diktovacím systému nebo v nástroji pro hlasové ovládání počítače, (ovládání navigace a nastavení robotů [4]), a samozřejmě ve výuce předmětů zabývajících se počítačovým zpracováním řeči. Distribuovaný systém potom v praxi využívala např. aplikace Infocity poskytující formou hlasové komunikace informace o dopravě, kultuře apod. [6]

Následující kapitola stručně popisuje princip zmiňovaného rozpoznávače, celého distribuovaného systému, dále potom uvádí, které faktory a do jaké míry ovlivňují jeho chod.

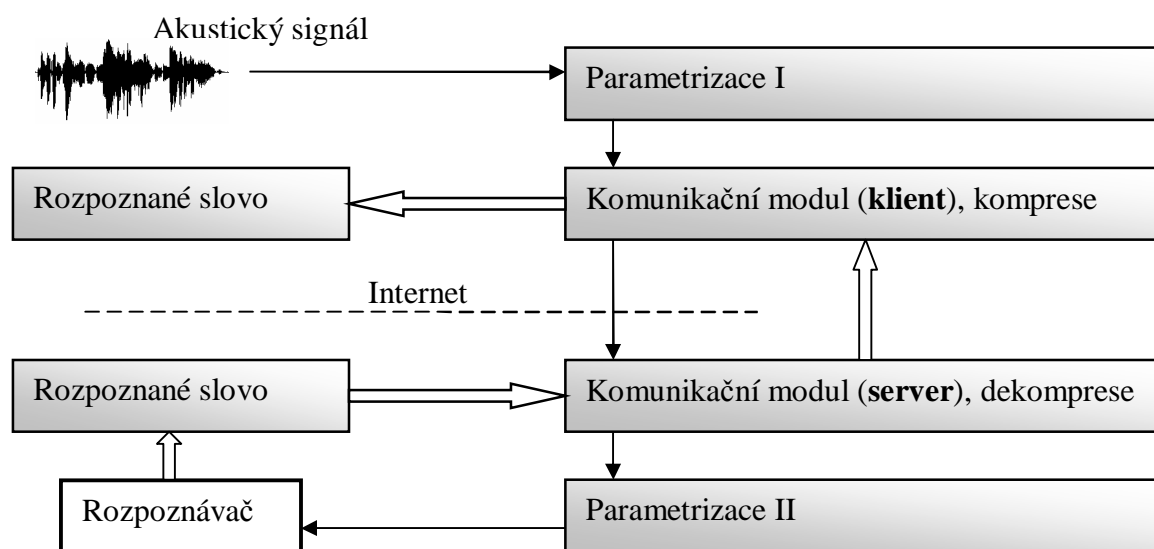
1.1 Funkce

Vzhledem k velké procesorové náročnosti v době vývoje zmiňovaného systému rozpoznávání vedla vývojáře myšlenka přesunout výpočetně náročné algoritmy na vzdálený hardware poskytovatele služby. V době uvedení systému do provozu byl výkon procesorových stanic z hlediska rozpoznávání řeči nedostatečný pro provoz v reálném čase. Distribuované rozpoznávání (Distributed Speech Recognition - DSR) přesouvá hlavní výpočetně náročný úkol rozpoznávání požadované fráze na vzdálený systém s mnohem vyšším výkonem. Tímto lze obejít hlavní nedostatek systému, na kterém běží klient – nedostatek procesorového výkonu pro rozpoznávač.

Systém je tedy postaven na architektuře klient – server, kde je využit TCP (Transmission Control Protocol) pro komunikaci po internetu mezi oběma stranami.

Při rozpoznávání slov systém vychází z uživatelského slovníku. Ten může obsahovat až stovky tisíc položek. Jeho velikost do jisté míry ovlivňuje úspěšnost rozpoznávání, která se může i více než 98% (v závislosti na velikosti slovníku, akustických podmínkách ve kterých se mluvčí nachází, hardwarové vybavení apod.) [1] a je tedy velmi vhodný pro použití v reálných situacích.

Rozpoznávač je založen na modelu skrytých Markovových modelů (HMM – Hidden Markov Models) a je schopný rozpoznávat izolovaná slova a krátké fráze na základě daného slovníku. Je nezávislý na mluvčím.



Obrázek 2: Architektura systému klient - server

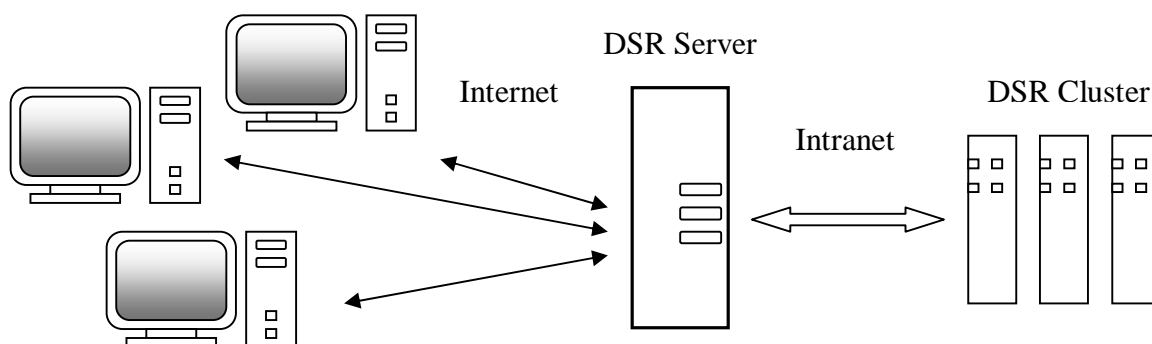
Na straně klienta probíhá pouze nahrávání a na výkon nenáročná parametrizace rozpoznávaného slova. Data jsou poslána přes internet na rozpoznávací server, kde proběhne hlavní výpočet. Záhy klient obdrží odpověď - rozpoznáný text, který v klientské aplikaci vyvolá příslušnou akci (Obrázek 2). V systému není vyžadován software třetích stran, což usnadňuje správu celého systému. Server umožňuje jak anonymní tak registrovaný přístup. Anonymní klient je omezen velikostí slovníku a délkou připojení. Na druhou stranu registrovaný uživatel je omezen pouze rychlostí rozpoznávacího serveru.

Architektura klient – server přináší mnoho výhod. Se serverem může komunikovat naráz více klientů, navíc každý může volit z několika druhů rozpoznávačů a rozpoznávacích modelů. Protože je hlavní část výpočtu přesunuta na

stranu serveru, mohou jinak procesorově náročný rozpoznávač využít i stanice s malým výkonem. To umožní návrh levných zařízení ovládaných hlasem. Pokud se navíc provozovatel rozhodne modifikovat rozpoznávač, může tak učinit, aniž by byl uživatel nucen stahovat novou verzi aplikace na svůj počítač. Nastavením rozdílných práv pro různé uživatele se zase může zajistit omezení pro určitou skupinu uživatelů (délka připojení, velikost a druh slovníku apod.)

Hlavní nevýhoda takto uspořádaného systému vyplývá z jeho podstaty – je vyžadováno trvalé připojení k internetu, což i přes jeho velkou rozšířenost nemusí být ve všech lokalitách možné. Spravování celého systému také klade jisté nároky na správce, kdy je vyžadována jistá znalost celé problematiky v oblasti rozpoznávačů.

Protože jsou nároky rozpoznávače při vyšším počtu připojených klientů k serveru vysoké, bylo navrženo použít síť pracovních stanic (network of workstations - NOW) místo jediného serveru (Obrázek 3). Doba odezvy klienta od serveru se tak dokáže udržet v takových mezích, aby i v případě provozu většího počtu připojených klientů nedocházelo k prodlevám. Toto řešení s sebou přináší další nároky na software, který musí dynamicky měnit parametry serveru podle počtu připojených stanic, rychlosti dotazů a být vybaven ochranou proti přetížení.



Obrázek 3: Topologické schéma víceuživatelského DSR systému s paralelním zpracováním

Průběh celého procesu rozpoznávání je následující: audio signál ze vstupního mikrofону je sejmут a zaznamenán klientskou aplikací. Po předzpracování a rozdělení signálu na jednotlivé rámce (framy) je signál parametrizován. Následuje přenos přes internet na server. Poté, co je zde slovo detekováno, je umístěno do fronty.

Rozpoznané slovo jde zpět po intranetu odesláno na server, který výsledek odešle v textové podobě klientské aplikaci. V tomto okamžiku je celý proces rozpoznání ukončen.

1.2 Reálný provoz

Aplikace využívající DSR vyžadují trvalé připojení do sítě internet. Nutným požadavkem je tedy stabilní, dostatečně rychlá linka, která zabezpečí přesun dat mezi klientem a serverem. Důležitým parametrem je také odezva od serveru. Na to, zda a jak bude systém použitelný má vliv celá řada dalších faktorů, které jsou ve stručnosti popsány dále.

V extrémních případech může nastat situace, kdy není možné vzhledem k velkému množství požadavků obsloužit všechny klienty a nastává pokles rychlosti odezvy. V těchto případech klesá tedy doba odezvy aplikace. Výzkum ukázal, že prodleva mezi promluvou a reakcí systému by neměla být delší než 1 - 2 sekundy (jak je dále uvedeno v kapitole 2), jinak dochází k poklesu pozornosti uživatele.[3] Systém tedy musí být navržen tak, aby tyto případné výkyvy byly vhodně ošetřeny.

1.2.1 Vliv internetového připojení

Jedním z faktorů, který může uživatel do jisté míry ovlivnit je kvalita použitého připojení k internetu. V současné době je v České republice dostupnost připojení i díky sítím mobilních operátorů velice dobrá. Zatímco před pár lety domácnostem dominovalo dial-up připojení přes telefonní linku s velice omezenou přenosovou rychlostí a velkou odezvou, nyní převládají řešení prostřednictvím bezdrátových sítí.

1.2.2 Vliv použitého vybavení

Špatný technický stav nebo nesprávné nastavení mikrofonu či zvukové karty. Při nesprávném nastavení hladiny hlasitosti se vstupní signál nenachází v požadovaných mezích, je tedy buď zkreslen vlivem přebuzení nebo naopak jeho síla není dostatečná na to, aby byl rozpoznávačem zaregistrován začátek promluvy. Signál, který získá klientská aplikace je tedy nepoužitelný a aplikace se tak jeví jako nefunkční.

Vliv zvukových karet (co se týče šumu a zkreslení) používaných v dnešní době (a v našem případě) je vzhledem k pokroku v technologii zanedbatelný a lze ho tedy vyloučit.

1.2.3 Vliv klientské aplikace

Uživatel nemůže ovlivnit to, jak byla daná aplikace naprogramována. Tvůrce musí dbát na správné řízení, časování a zobrazování dialogů, důležitá je i zpětná vazba, která poskytuje uživateli přehled, jak systém reaguje. reakce aplikace, pokud je od uživatele při Podrobnosti jsou uvedeny v kapitole 2.

1.2.4 Vliv okolního prostředí

U tohoto testovaného rozpoznávače se předpokládá provoz v relativně klidném prostředí např. domácnosti nebo kanceláři. Vzhledem k rozšíření bezdrátového připojení k internetu je možný běh aplikací využívající tento DSR server mimo budovy, i v dopravních prostředcích. Hladina okolního hluku může vrst nad mez, kdy již není možné odlišit promluvy uživatele od hluku. Úspěšnost rozpoznávání prudce klesá. Řešením je změna polohy mikrofону, případně změna stanoviště.

1.2.5 Vliv způsobu promluvy uživatele

Hlas každého řečníka je unikátní, navíc se v průběhu sezení může měnit. Do barvy a charakteru hlasu se promítá věk, nálada, emoce nebo nemoc. Uživatel může mít také vadu řeči nebo mluvit s určitým přízvukem či nářečím. Slovo může vyslovit pokaždé jiným tempem.

Ovlivnit úspěšnost rozpoznávače mohou tedy, kromě samotného rozpoznávacího systému i velké množství ostatních faktorů. I přes to se daří při rozumné velikosti a skladbě slovníku dosahovat velice dobrého rozpoznávacího skóre.

2 Klientské aplikace

Klientská aplikace v tomto případě zajišťuje získání promluvy od uživatele, její zpracování a předání serverové části, od které získá výsledek v podobě rozpoznání slova, fráze nebo věty. Obecně může být vybavena grafickým rozhraním v případě běhu na osobních počítačích, mobilních zařízeních typu PDA apod. nebo pouze hlasovým vstupem a výstupem v případě dialogových systémů provozovaných např. po telefonu. Následující kapitola se nejprve zabývá obecnými pravidly při návrhu uživatelského rozhraní interakce člověka s počítačem. Následně jsou zmíněna pravidla, která je vhodné dodržovat při návrhu aplikace ovládané hlasem.

2.1 Obecná pravidla

Pokud se nejdříve zaměříme na obecné principy návrhu, dříve či později zjistíme, že existují určitá pravidla, která, pokud jsou dodržována, mohou zásadně ovlivnit uživatelský pohled na aplikaci. Ben Shneiderman, odborník zabývající se interakcí člověka s počítačem uvádí osm pravidel (Eight Golden Rules), kterých je vhodné se držet při návrhu rozhraní [19]:

Konzistence – zahrnuje jak konzistentní vzhled (podobná dialogová okna, fonty, barevná schémata), stejnou terminologii na různých místech aplikace, tak i podobné chování v podobných situacích

Umožnit použití zkratk – spadají sem makra, klávesové zkratky, ikony důležitých funkcí na rychle přístupných místech a jiné

Zpětná vazba – systém by měl vždy reagovat na akci uživatele nějakou (pro konkrétní situaci přiměřenou) odezvou

Organizace akcí do uzavřených celků – tj. komplikované akce rozdělit do menších celků, tak aby uživatel mohl přehledně pomocí zpětné vazby sledovat průběh od začátku do konce

Prevence chyb a řešení chybových situací – v první řadě by se mělo chybám předcházet; v případě, že chyba nastane, naznačit uživateli postup pro její odstranění

Možnost vrátit akci zpět - v aplikacích označováno termínem „undo“

Poskytnou plnou kontrolu pro zkušené uživatele – aplikace se nesmí chovat nepředvídatelně a umožnit použití pokročilých funkcí co nejjednodušeji; uživatel by měl mít dojem, že aplikaci ovládá on a ne naopak

Nepřetěžovat krátkodobou paměť uživatele – seznamy a struktury různých menu by měli obsahovat přiměřený počet položek

Tyto body jsou dále podrobněji rozebrány ve vztahu k hlasovým rozhraním v dalších odstavcích.

Obecně samotná přesnost rozpoznávačů, která se s postupným vývojem zlepšuje, ještě sama o sobě nezajistí použitelnost aplikace. Tabulka 1 zobrazuje různé kategorie omezení, které se při návrhu musí vzít v úvahu [7]

Tabulka 1: Omezení týkající se hlasových rozhraní

Kategorie	Definice	Specifická omezení
Rozpoznávač řeči	Omezení současných rozpoznávačů řeči	Chyby, konečná velikost slovníku, jazykový model, akustický model, kvalita vstupního signálu
Mluvená řeč	Omezení spojená s řečí a mluveným jazykem	Spontánní promluva, obecná povaha, nejednoznačnost, omezená schopnost vyjádření se, souvislosti, systém střídání jednotlivých mluvčích
Prostředí	Vliv prostředí na uživatele	Hluk, přerušování, více mluvčích v jednom okamžiku
Lidské vědomí	Vlastnosti kognitivního systému	Omezená kapacita paměti, malá vytrvalost, problémy s doslovným zpracováním
Uživatel	Odlišnosti a preference mezi uživateli	Předchozí zkušenosti a očekávání, motivace, omezená znalost úlohy
Hardware	Vlastnosti hardwaru	Propustnost linky, mikrofon, CPU

2.1 Využití aplikací

Řeč je vhodná pro určité aplikace více, pro jiné méně. Kde je například myš jako polohovací zařízení vhodnější než hlas, jednotlivé stisky kláves nebo jejich kombinace lze s přehledem hlasovým vstupem nahradit. Rozhraní by měla být navržena s ohledem na uživatele, aby vyřešení úkoly bylo co nejjednodušší, pohodlné a styl práce ho neodradil od dalšího používání.

V další části jsou zmiňovány aplikace, které se dotýkají tématu této práce, tj. pracující v režimu rozpoznávání izolovaných slov. Mohou být rozděleny na:

- desktopové aplikace
- mobilní aplikace

2.1.1 Desktopové aplikace

Jedním z nejstarších prototypů pro rozpoznávání řeči v prostředí osobních počítačů jsou povelové a řídicí aplikace (command and control), které jsou zde používány hlavně pro navigaci operačními systémy a aplikacemi, které pod nimi běží. Od uživatele aplikace očekává slovo nebo krátkou frázi z předem daného seznamu ve slovníku, které systém porovnává s konkrétní promluvou. Příkazy se mohou skládat a vytvářet tak velké množství kombinací (např. spojením předem daného povelu a číslovky: „Přejít na stránku {číslo}”). Aplikace však nemusí reagovat pouze na pevně dané slova ze slovníku, ale hledat klíčová slova v promluvě uživatele (word spotting). Například povely „skoč na další“ a „přejdi na další“, tak vyvolají stejnou odezvu.

Možnosti využití tohoto konceptu je spousta, např:

- odpověď na systémový dotaz (např. ano/ne v dialogových oknech)
- aktivace maker
- editace dokumentů
- výběr ze seznamů
- jednoduchý dialog

Dalšími možnostmi jsou programy umožňující diktování izolovaných slov, průchody rozsáhlými menu nebo sloužící pouze jako doplňky k již existujícím aplikacím se standardním vsupem.

Předem definované povely by měli být pro uživatele intuitivní a ne příliš dlouhé. Zobrazení seznamu aktuálně použitelných příkazů na obrazovce dává uživateli okamžitou nápovědu - uživatel pak přirozeně vyslovuje text, který vidí na obrazovce.

2.1.2 Mobilní aplikace

Malá mobilních zařízení postrádají klasické vstupní prvky jako klávesnici a myš známé z ovládání desktopových aplikací. Ovládání pomocí řeči se tedy přímo nabízí. Limitem je zde procesorový výkon. Vzhledem k tomu, že lze mobilní zařízení provozovat kdekoli, je nutné brát v úvahu i otázku soukromí (použití na místech s možností přímého odposlechu – obchody, parky, hromadná doprava a další).

2.2 Velikost a struktura slovníku

Použitím slovníku o malém počtu slov (např. méně než 50 slov) docílíme větší přesnosti klasifikace slova než se slovníkem čítajícím tisíce položek, malý slovník ale omezuje uživatele počtem voleb, které může použít. Také to někdy znamená prodloužení dialogu o další kroky. Slova pro příkazy určité skupiny by měla být dostatečně fonologicky odlišná, aby nedocházelo k chybné klasifikaci.

Tabulka 2 ilustruje na příkladu bankovního systému ovládaného hlasem možné styly dialogu a jejich vztah k velikosti slovníku. Režim „1“ pracuje s největším slovníkem, režim 4 s nejmenším. Jak se velikost slovníku snižuje, roste pravděpodobnost správné klasifikace slova.

Tabulka 2: Režimy hlasového vstupu

	Režim	Příklad
1	Dotaz	Jakou službu si přejete aktivovat?
2	Menu	Zvolte službu; zůstatek, bankovní převod nebo jinou?
3	Ano/Ne	Přejete si znát zůstatek na Vašem účtu?
4	Zvuk	Pokud chcete slyšet zůstatek na Vašem účtu, vydejte zvuk

V praxi se proto využívá strukturování slovníku na soubory slov, která se mohou objevit v daném stavu dialogu nebo která mohou následovat nějaké bezprostředně vyslovené slovo [5]

2.3 Řízení dialogu

Budoucností v technikách dialogu mezi člověkem a počítačem je interakce prostřednictvím přirozeného jazyka. Uživatel velice jednoduše zadá příkazy běžnými formulacemi přirozeného jazyka a povede tak dialog podle svých úvah. V současné době je však tato možnost stále ve vývoji.

V případě dialogu po telefonu je, kromě tónové volby, pro člověka řeč jediná možnost komunikace se systémem po telefonu. Důležité je volajícího v prvním kroku upozornit na to, že mluví s automatizovaným systémem a ne s člověkem. Nejfrekventovanější položky by měly být obsaženy v menu nejvyšší úrovně. Běžný uživatel je schopný si zapamatovat najednou tři až čtyři položky.

Rychlost, kterou lze získat informaci pomocí řeči je nižší než při získávání vizuální informace. Zatímco v grafickém prostředí můžeme mít všechny informace zobrazené v jednom okamžiku, u řeči, díky své sériové podstatě informace to neplatí.

Řízení je možné rozdělit následovně:

- dialog řízený systémem
- dialog řízený uživatelem
- kombinované řízení

V prvním případě je uživatel veden počítačem na základě otázek jednotlivými volbami. Dialog řízený uživatelem většinou využívá vyhledání a identifikaci klíčových slov v rozpoznaném textu a podle toho vymezuje okruh, ve kterém se bude dialog nadále pohybovat.

Zde jsou uvedena některá z pravidel konstrukce dialogu [2]

- V případě, kdy systém vyjmenovává aktuální volby a není možné přerušení řídicím povelom od uživatele by tyto seznamy měli obsahovat maximálně 5 položek (více položek by se mělo rozvětvit do několika skupin). Pokud je povolen režim barge-in, je možné počet položek rozšířit až na 9.
- Výzvy aplikace by měly být krátké, zejména uvítací hlášení.
- Zpětná vazba je nezbytná, neměla by však zdržovat neustálým opakováním uživatelova hlasového vstupu

- Dialog má být navržen co nejpřirozeněji, s ohledem na rovnoměrné střídání uživatele a systému, nespolehat se pouze na možnost přerušení systému uživatelem (barge-in)
- Při chybě nebo vypršení časového limitu v rámci nápovědy by měl být uveden příklad správného vstupu
- Dotazy vyžadující pouze odpověď Ano/Ne jsou velmi robustní a dokáží spolehlivě vyvést dialog v případě chyb a nejasností

Pokud je aplikace schopna uchovat informace o uživateli, může se přístup k novému uživateli nebo zkušenému uživateli lišit. Nováčkům, kteří nejsou se systémem seznámeni, je nabídnuta větší míra nápovědy než uživatelům, kteří jsou s aplikací dobře seznámeni.

Při návrhu je také třeba kontrolovat prodlevy v komunikaci. Pokud nastane delší pauza ze strany systému, systém tím naznačuje, že je na řadě uživatel.

2.4 Detekce a korekce chyb

Pomineme-li chyby aplikace způsobené samotným programovým kódem, může nastat množství chybových stavů, kde velkou roli hraje samotný rozpoznávací systém. Vzhledem k tomu, že rozpoznávání hlasu není dokonalé, zasahují události, kdy je slovo nesprávně rozpoznáno citelně do běhu aplikace. U programu ovládaného standardním vstupem (klávesnicí a myší) je frekvence chyb pro stisknutí tlačítka nebo klávesy kolem 5%. Pokud se uživatel dopustí překlepu nebo se překlíkne v tomto systému, bere to jako chybu, kterou zavinil on sám. U aplikací ovládaných hlasem však jakékoliv chybné rozpoznání vnímá jako chybu samotného systému.[9] Předcházení chybám musí být proto věnována velká pozornost. Vhodným řešením při řízení chyb lze zachránit i jinak ne zcela přesný rozpoznávač. V oblasti detekce a řízení chyb lze uplatnit následující schéma:

- 1) detekce
- 2) diagnóza a příčiny
- 3) plán korekce
- 4) vykonání korekce
- 5) informování uživatele
- 6) návrat

7) (prevence)

V aplikacích s rozpoznávače je možné se setkat s následujícími druhy chyb:

Systém nerozezná začátek promluvy

Model rozpoznávače nerozezná, kdy uživatel vyslovil frázi. Aplikace se tak chová, jako by mluvčí nic nevyslovil a jeví se v danou chvíli jako nefunkční. Aplikace by měla být ve stálém kontaktu s mluvčím a nějakým způsobem mu dát najevo, že systém naslouchá (např. ikonou v grafickém rozhraní nebo hlasovou výzvou). Je možné použít i kameru, která snímá obličej uživatele a pomoci tím tak k vyhodnocení, kdy uživatel promluvil.

Zamítnutí slova

Rozpoznávač sice zachytí začátek promluvy, ale nemůže rozhodnout, jaké slovo bylo vyřčeno. Na vině může být silný okolní hluk, kdy promluva splyne s okolím nebo když se daná fráze nenachází ve slovníku. Systém proto požádá uživatele o zopakování slova. Frustrující je, pokud se tato chyba vyskytne několikrát za sebou, proto je důležité eliminovat nadměrné opakování výzvy systému a uživatele postupně navést tak, aby svoji volbu specifikoval jinak. Uživatel má také při opakování nerozpoznané fráze tendenci přehnaně artikulovat, což může mít za následek naopak zhoršené podmínky pro rozpoznávač.

Záměna slova za jiné

Rozpoznávač určí vyslovené slovo jako jiné, často podobné původnímu slovu („červená“ – „černá“). Řešením je systém potvrzujících otázek („Opravdu ukončit spojení?“), který ale nelze využívat stále, protože při nadměrných výzvách ze strany systému by docházelo ke zdržování uživatele. Dále je nutné omezování slovníku pouze na ta slova, která jsou v dané chvíli relevantní a vyvarování se používání podobně znějících slov pro aktivaci rozdílných úkolů.

Klasifikace hluku jako platné slovo

Na vině je silný hluk, který překrývá promluvu, úder do mikrofonu apod. Takováto událost je poté buď zamítnuta nebo interpretována jako platné slovo a systém nějakým způsobem reaguje, aniž by mluvčí vyslovil frázi. Je proto vhodné pokud možno vymezit intervaly, kdy aplikace naslouchá (případně použít kameru, jak je uvedeno výše) a snížit tím tak možnost omylu.

2.5 Zpětná vazba

Zpětná vazba je prostředek umožňující jak informování uživatele o stavu aplikace, tak i předcházející množství chybových stavů, do kterých by se jinak systém mohl dostat.

Zpětnou vazbu rozdělit na primární a sekundární. Primární zpětnou vazbu lze využít, když rozpoznané promluvy vyvolají pozorovatelné akce a případné chybné klasifikace nezpůsobují bezprostřední nenapravitelné škody a lze je rychle a bez větších ztrát korigovat. Typickým příkladem je zobrazením výsledků klasifikace ještě před provedením požadovaného zásahu [5]

Uživatel by měl být informován zejména v těchto situacích:

- potvrzení akce
- systém je zaneprázdněn
- akce nemůže být vykonána
- čeká se na promluvu uživatele (systém naslouchá)
- nesprávná úroveň vstupního signálu z mikrofону

Sekundární zpětnou vazbou systém informuje o výsledku klasifikace. Uživateli je potom většinou dána možnost buď výsledek potvrdit nebo zamítnout a opravit. Pokud neexistuje možnost vizuální kontroly klasifikace, měl by se tento typ vazby použít. Podle Jakoba Nielsena, odborníka na použitelnost softwarových aplikací lze rozdělit dobu odezvy systému rozdělit na několik částí.

V [12] jsou uváděny tyto časové limity a jejich vnímání uživatelem:

- **0.1 sekundy:** je limit, kdy uživatel vnímá, že systém reaguje okamžitě
- **1 sekunda:** uživatel již prodlevu zaznamená
- **10 sekund:** do tohoto limitu je uživatel schopen koncentrovat se na zpracování úkolu; zde je již zpětná vazba nezbytná, aby uživatel věděl, že systém zpracovává požadavek

Pokud rozhraní nedisponuje grafickým výstupem, prodleva mezi vyslovením příkazu a reakcí systému by neměla být delší než 2-3 sekundy. [9] uvádí, že prodleva kratší než 2 – 3 sekundy je akceptovatelná i bez zpětné vazby.

Pokud je zpětná vazba poskytována prostřednictvím syntezátoru, je hlavním požadavkem dostatečná srozumitelnost. Řeč ve formě nahrávky lidské řeči má stále nejlepší kvalitu, ale v porovnání s TTS je neflexibilní[5]

3 Názor uživatele

Dobře naprogramovaná aplikace ještě nezaručí to, že se s ní uživatelům bude dobře pracovat. Je proto nutné zjistit, jak uživatel vnímá práci s danou aplikací a podle toho upravit rozhraní a chování aplikace tak, aby nejlépe odpovídala požadavkům koncových uživatelů. Může se postupovat těmito způsoby:

- zjistit potřeby uživatele a podle toho navrhnout aplikaci
- navrhnout aplikaci a nechat ji uživatelem zhodnotit

Hodnotit lze aplikaci jako celek nebo pouze dílčí části systému. Lze tedy hodnotit například vzhled, odezvu aplikace, srozumitelnost syntetizovaného hlasu [18] a spoustu dalších prvků, které jsou důležité pro to, aby měl uživatele dobrý pocit při práci s aplikací.

Měřítko celkové úrovně aplikace mohou být objektivní nebo subjektivní:

Objektivní měřítko

Obvykle se měří čas splnění daného úkolu, počet kroků k dokončení úkolu nebo míra dokončení úkolu. Dále se může brát v úvahu např. počet kroků potřebný k ošetření chyb a uvedení aplikace do normálního režimu.

Subjektivní měřítko

Do této kategorie spadá celkový pohled uživatele na testovanou aplikaci, tj. jak ji vnímá, jak je celkově s danou aplikací při práci spokojen.

Jedním z velice důležitých faktorů, které ovlivňují použitelnost aplikace je četnost chyb. K tomu, aby se nějakým způsobem dalo posoudit, zda je uživatel s aplikací spokojen bylo vytvořeno velké množství metod, pomocí kterých lze určit, zda je aplikace použitelná v reálném provozu a jak ji běžný uživatel vnímá.

Při hodnoceních uživatele aplikace, setkáme se s definicí pojmu použitelnosti a dalších souvisejících pojmů. Definice mohou znít například následovně [13]:

Použitelnost: Efektivnost, účinnost a spokojenost se kterou daný uživatel dosáhne daných cílů v určitém prostředí.

Přičemž pojmy efektivnost, účinnost, spokojenost mohou být definovány takto:

Efektivnost (angl. effectiveness) – je přesnost a úplnost s kterou uživatelé může dosáhnout těchto cílů

Účinnost (angl. efficiency) - může být definována jako námaha k dosažení cílů. Účinnost v podstatě představuje, jak snadno uživatelé mohou program použít ke svému zamýšlenému účelu.

Spokojenost (satisfaction) - vztahuje se ke komfortu, uživatelskému vnímání, pocitům a názorům.

Zatímco efektivnost a účinnost se vztahují k objektivním měřítkům (např. rychlost splnění daného úkolu), spokojenost je čistě subjektivní dojem uživatele.

Metod hodnocení software z hlediska použitelnosti a kvality je celá řada a lze je samozřejmě rozdílně třídit podle různých kritérií. Následují stručné příklady některých přístupů k dané problematice.

3.1 Hodnocení odborníky

Skupina odborníků může zahrnovat experty na použitelnost, softwarové vývojáře nebo další odborníky orientující se v dané problematice [16]:

- heuristické hodnocení – jedná se o nezávislé ohodnocení rozhraní několika odborníky (obvykle 3 až 5) a jejich následné porovnání se stanovenými principy použitelnosti, které se nazývají heuristiky
- kognitivní procházení – zahrnuje procházení odborníka cílem testování a provádění vybraných úkolů pro zjištění, zda budou pro uživatele srozumitelné a jednoduché

3.2 Hodnocení uživateli

Zde jsou do skupiny hodnotitelů vybírány běžní uživatelé. Výběr může být zcela náhodný či s přihlédnutím na cílovou skupinu [11]

- uživatelské testování - uživatelé jsou vyzváni k vypracování předem stanovených úkolů a jsou monitorováni testující osobou, která zaznamenává vzniklé problémy při samotném testu
- trénovací metoda – účastníci tohoto testu se mohou dotazovat přítomného experta/testujícího na otázky související s hodnoceným systémem. Účelem je tedy odhalit jaké informace uživatelé požadují při používání systému

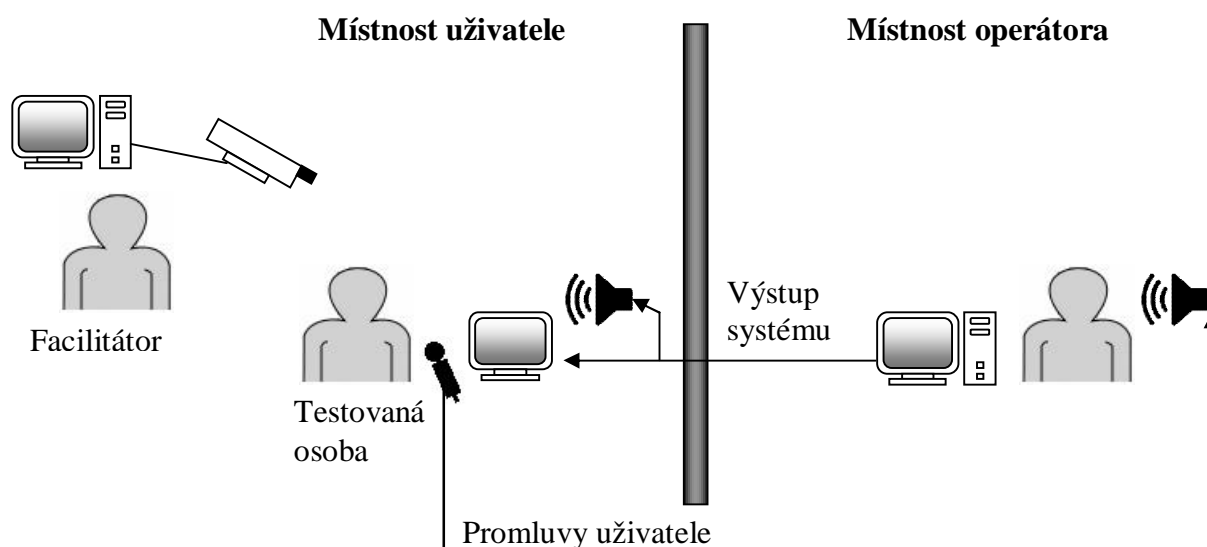
- dotazník
- rozhovor

3.3 Prototyp

Je-li třeba zjistit názor uživatele na technologii, která ještě nebyla vyvinuta, je vhodné použít některou z metod, využívající prototypu určitého systému.

3.3.1 Wizard of Oz

Termínem Wizard of Oz (WoZ) je označována metoda pro testování ještě neexistujících systémů, či funkcí systému, které ještě nebyly implementovány. Funkce, které v systému chybí zastává vyškolená osoba (wizard) nebo celý tým a simuluje tak některé nebo všechny odezvy systému. O tom však uživatel není informován, domnívá se tedy, že pracuje s počítačově ovládaným systémem. Tato technika je často používána při testech dialogových systémů, aplikací ovládaných hlasem. Rozpoznávač řeči zde může být zastoupen člověkem a lze tak nasimulovat např. bezchybné rozpoznávání řeči. Schéma je na obrázku 4.



Obrázek 4: Schéma metody Wizard of Oz [8]

Cílem je studium chování člověka při interakci s počítačovým systémem. Osoba zastupující roli operátora musí být velice dobře vyškolená, aby výstup, který ovládá, odpovídal výstupu počítače.

Výhodou je jistě možnost otestovat technologii, která ještě neexistuje či zhodnotit systém již na počátku vývoje předtím, než se rozhodne, jaké funkce budou zahrnuty v konečném produktu. Také je možné velmi pružně měnit reakce systému při přípravě dalšího sezení.

Nevýhodou je náročný trénink operátora v roli wizard, která musí podávat během sezení stálý a věrohodný výkon, nedopouštět se překlepů a řídit celý systém tak, aby nebyl odhalen.

Celkově je tato metoda tedy dosti finančně náročná.

3.4 Dotazník

K získání informací od uživatelů se osvědčila metoda využívající dotazníků. Dotazník slouží ke sběru zejména dat kvantitativních, ve kterém má hodnotitel na jisté škále ohodnotit jeho dojmy a pocity, které v něm používání aplikace vyvolalo. Může se jednat o dotazník jak v papírové formě, tak i ve verzi pro osobní počítač (online nebo offline)

Existují různé typy hodnotících škál a také různé výstupy z takovýmto způsobem sesbíraných dat.

Výhodou metod využívající dotazníků jsou nízké náklady. Pokud jsou otázky pečlivě pokládány, nepotřebuje testovaná osoba asistenci. Dalším kladem je možnost oslovení velkého počtu lidí díky volbě vyplnění v elektronické formě a odeslání dat přes internet. Sběr dat lze tak částečně zautomatizovat.

Nevýhodou jsou naopak ne vždy zcela přesná data, způsobená například tím, že uživatel neuvede pravdu nebo že se není schopen rozhodnout vzhledem k omezenému rozsahu hodnotící škály.

3.4.1 Výběr a skladba otázek

Klíčovým prvkem je výběr a struktura dotazů. V dotazníku by se měly objevit pouze dotazy, které bude moci uživatel s jistotou odpovědět. Dále se nesmí vyskytovat otázky podobného znění. Délka dotazníku nesmí přesáhnout únosnou mez, protože většina uživatelů nebude jistě nadšená z dotazníku čítající např. deset stran.

3.4.2 Příklad dotazníku

Následuje příklad části dotazníku pro hodnocení použitelnosti počítačového systému s pětibodovou stupnicí (obrázek 5)

	Silný nesouhlas			Silný souhlas	
1. Myslím, že bych tento systém používal často					
	1	2	3	4	5
2. Systém se mi zdá zbytečně komplikovaný					
	1	2	3	4	5
3. Myslím, že systém se používá snadno					
	1	2	3	4	5
4. Myslím, že bych potřeboval asistenci kvalifikované osoby, abych mohl používat tento systém					
	1	2	3	4	5
5. Rozmanité funkce jsou v systému dobře začleněny					
	1	2	3	4	5
6. Myslím, že v tomto systému je mnoho rozporuplností					
	1	2	3	4	5
7. Dokážu si představit, že většina lidí se naučí s tímto systémem pracovat velice rychle					
	1	2	3	4	5
8. Systém se mi zdá velice těžkopádný					
	1	2	3	4	5
9. Při používání systému jsem se cítil velmi sebejistě					
	1	2	3	4	5
10. Musel jsem se naučit mnoho věcí, než jsem dosáhl nějakých výsledků					
	1	2	3	4	5

Obrázek 5 Příklad dotazníku uvedeném v [14]

3.5 Výběr hodnotitelů

Získávání uživatelů pro hodnocení je další podstatný krok pro úspěšný sběr dat. Výběr vzorku populace by měl přibližně odpovídat cílové skupině, pro kterou je konkrétní software vyvíjen. Jejich počet by měl být dostatečný, při menších počtech by se metoda mohla vykazat nepřesné výsledky. Vše se odvíjí od velikosti celého projektu a použité metody testování.

V případě velkého projektu se k oslovení veřejnosti osvědčila reklama v podnikových (školních) žurnálech, místních novinách, přímé telefonáty do vybraných organizací, rozdávání dotazníků, předvedení ukázkového videa nebo hierarchický systém, kdy mluvčí, který zapojí do projektu určitý počet dalších dobrovolníků získá speciální odměnu.

Testovací nahrávky by měly vznikat v prostředích a situacích, které budou při běhu reálné aplikaci nastávat, nesmí samozřejmě dojít k ohrožení všech zúčastněných osob v průběhu testů (např. čtení novin při řízení automobilu, nahrávání v neprozkoumaných nebezpečných místech apod.)

3.6 Příklady existujících metod

Následují příklady dvou konkrétních hodnotících metod využívající dotazník pro sběr dat od uživatelů. Zatímco metoda SUMI je vyvinuta pro obecná rozhraní, metodu SUXES je určena hlavně pro řečové a multimodální aplikace.

3.6.1 SUMI

SUMI (Software Usability Measurement Inventory) je pečlivě testovaná a prověřená metoda měření kvality softwaru z pohledu koncového uživatele. Metoda je doporučena jakékoliv organizaci, která si přeje určit míru použitelnosti vyvinuté aplikace.[15] Je dostupná v různých jazykových mutacích.

Hodnocení probíhá na základě dotazníku, který obsahuje 50 tvrzení. Uživatel u každé položky na základě svého rozhodnutí označí možnost: Souhlasím, Nevím, Nesouhlasím.

Tabulka 3: Příklady tvrzení v dotazníku SUMI

č.	Znění
1.	Software reaguje na vstup příliš pomalu
3.	Instrukce a výzvy jsou užitečné
13.	Způsob, jakým systém prezentuje informace je jasný a srozumitelný
22.	Nechtěl/a bych používat tento software denně

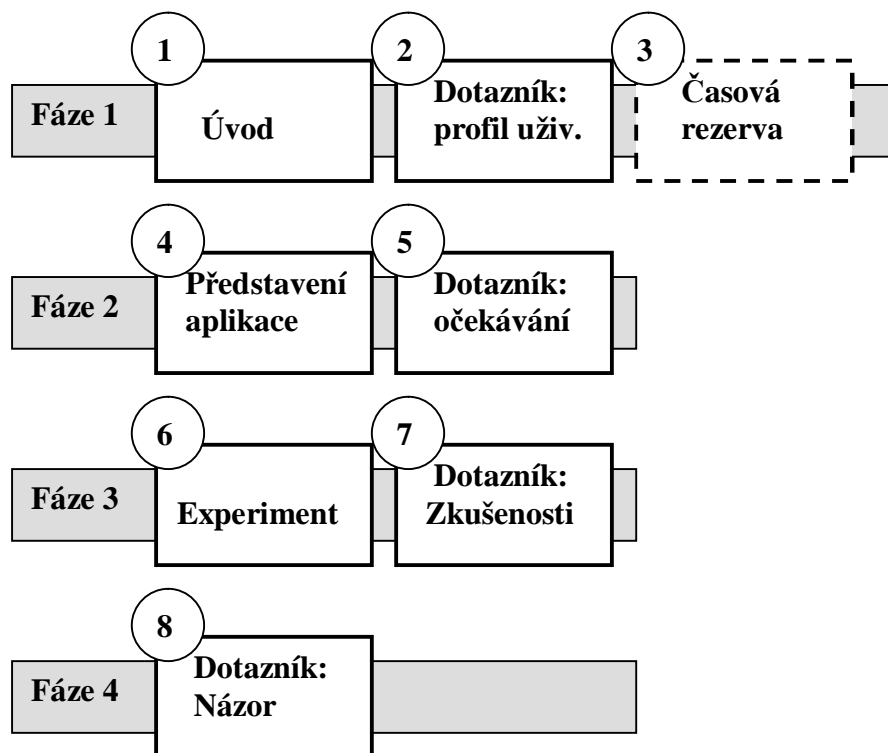
Vyplnění dotazníku trvá zhruba 3 minuty. Vedle klasického papírového dotazníku existuje i možnost vyplnění on-line verze.

Doporučuje se skupina nejméně třiceti hodnotitelů, ale dostatečné výsledky je možné získat i při počtu dvanácti hodnotitelů. Testy byly prováděny i na vzorku 3 nebo 4 uživatelů. Pokud je hodnotitelů méně, sebraná data již nepodávají tak přesnou informaci a k výsledkům je třeba přistupovat opatrně. Proto je doporučeno k testům přizvat co nejvíce reprezentativních uživatelů.

Vyplněné dotazníky jsou poté zaslány ke zpracování. Výsledky přehledně zobrazují skóre pro jednotlivé položky (účinnost, naučitelnost, apod.)

3.6.2 SUXES

SUXES je hodnotící metoda pro sloučení subjektivních metrik s uživatelskými experimenty. Pokrývá jak oblast očekávání, tak i zkušenosti uživatelů. Tím je umožněno analyzovat stav aplikace, její interakci s uživatelem a porovnávat výsledky. Je určena pro aplikace multimodální a hlasem ovládané. Použita je sada dotazníků před a po používání aplikace.



Obrázek 6: Hodnotící procedura metody SUXES [13]

Tak je možné určit rozdíl mezi očekáváním a zkušeností s danou aplikací. Celý proces je řízen webovým průvodcem a tak je dosaženo částečné automatizace.

Procedura je rozdělena do čtyř fází, jak ukazuje obrázek 6

Fáze 1: Uživatelé jsou představeny cíle metody. Dále jsou v dotazníku získány základní údaje (věk, jak často používá software ovládaný hlasem, apod.)

Fáze 2: Slouží k představení dané aplikace a její možnosti z hlediska vstupu a výstupu bez konkrétnějších detailů. Uživatel vyplní dotazník na základě tohoto představení.

Fáze 3: V této fázi uživatel pracuje s danou aplikací a plní zadané úkoly. Poté vyplní dotazník, který obsahuje stejná tvrzení jako v bodě 5 (obrázek x)

Fáze 4: V konečném kroku vyplní další dotazník, který je obvykle pro každou aplikaci jiný. Následovat může také rozhovor.

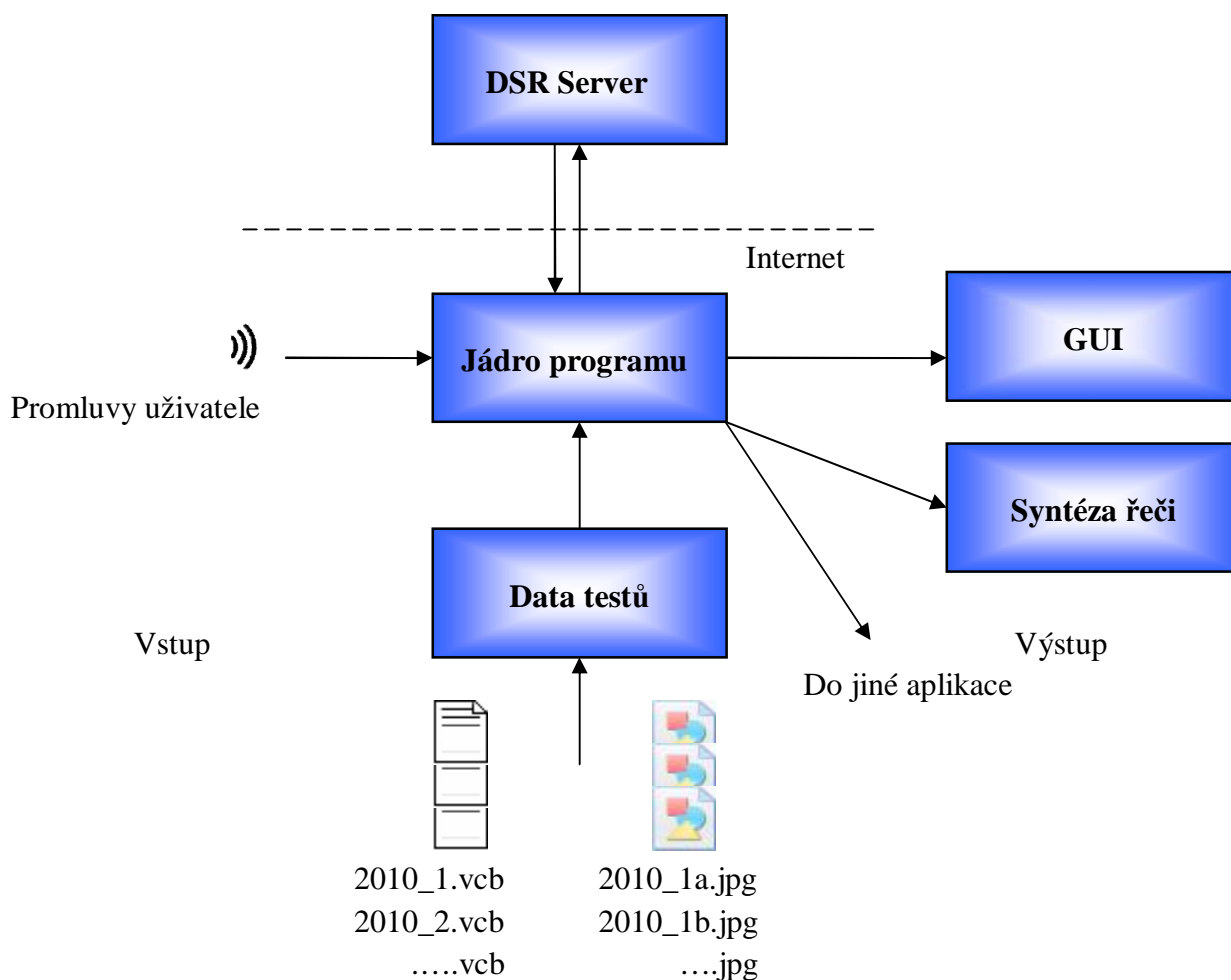
Důležitý je zde tedy rozdíl mezi tím, co uživatel od aplikace očekává a do jaké míry jeho očekávání aplikace splnila. To je zohledněno i v celkovém závěrečném hodnocení.

4 Testovací aplikace

Pro potřeby testů byla navržena klientská aplikace běžící pod systémem Microsoft Windows. Prvky pro nastavování testů a komunikace se serverem lze ovládat pouze myší (klávesnicí), naopak testy pouze hlasem. Aplikace využívá a rozšiřuje stávající jádro. Jako vývojové prostředí bylo použito Microsoft Visual Studio 2008. Program byl vytvořen v jazyce C++ na platformě .NET 3.5. Pro běh program tedy vyžaduje nainstalovaný .NET Framework 3.5. Dále je nutné připojení k internetu, sluchátka, mikrofon nebo headset.

4.1 Koncepce programu

Struktura aplikace je na obrázku 7.



Obrázek 7: Struktura testovací aplikace

Jádro programu zahrnuje získání promluvy uživatele, načítání a řízení testů a komunikaci se serverovou částí. Výstup je možný jak prostřednictvím grafického rozhraní, tak pomocí syntezátoru. Data pro jednotlivé testy jsou získávána prostřednictvím .vcb souborů, uložených v adresáři aplikace. Slovníky mohou mít velikost přibližně do 400 položek. Zároveň některé testy používají zobrazování obrázků formátu jpeg, uložených v adresáři obr.

4.1.1 VCB soubor

Tento soubor definuje jednotlivé slovníky; obsahuje slovník vždy pro každý konkrétní test. Jedná se o běžný textový soubor. Slovník je zapisován vždy do dvou sloupců podle schématu:

"slovo vypsané na obrazovku" "očekávané slovo"

Pokud jsou obě slova stejná, je v horním panelu vypsán stejný text. Pokud se slova liší, je po rozpoznání slova ve slovníku zobrazeno do panelu slovo v levém sloupci slovníku, tedy např.:

„Monday“ „pondělí“

vypíše do pole „Monday“

Kromě toho lze použít speciálních značek. Pokud je rozpoznáno slovo k nim přiřazené, vykonají funkci podle tabulky 4

Tabulka 4: Seznam značek a jejich význam

Značka	Význam
" "	ohraničuje text
EXE	spustí aplikaci definovanou parametrem
#PIC	zobrazí obrázek
#ROB	pohyblivý obrázek
#SAY	syntéza řeči v reakci na promluvu
//	komentář

Spuštění externí aplikace

Po rozpoznání promluvy spustí aplikaci specifikovanou cestou k exe souboru.

Zobrazení obrázku

Zobrazí obrázek přiřazený dané frázi v pravém panelu hlavního okna (viz kapitola Grafické rozhraní)

Pohyblivý obraz

Zobrazí okno s nastaveným obrázkem, kterým je možno pohybovat ve čtyřech směrech po obrazovce.

Promluva syntazátoru

Je možné vyvolat promluvu pomocí syntezátoru, jako reakci na promluvu mluvčího.

4.1.2 Režimy aplikace

Aplikace umožňuje pracovat ve dvou režimech:

- Normální
- Prezentace

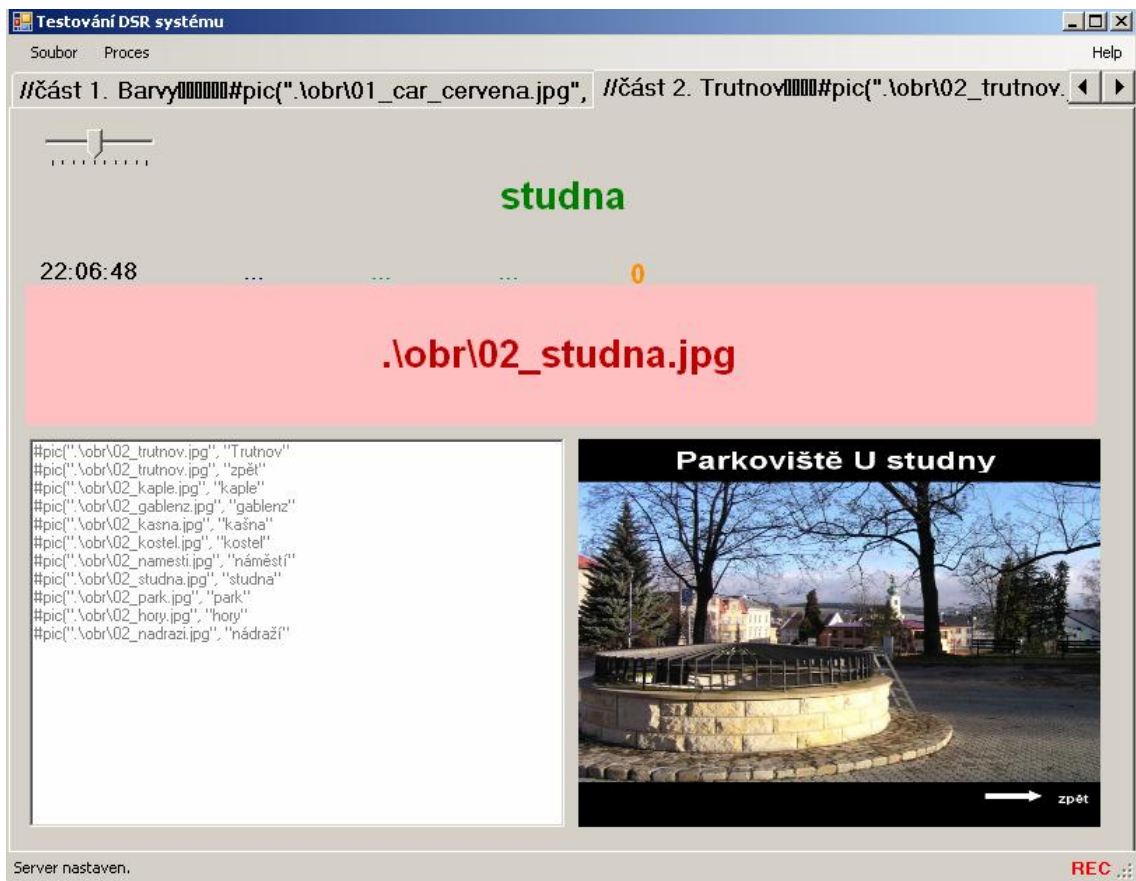
V normálním režimu aplikace čeká na hlasový vstup uživatele.

V režimu prezentace (je zvolena tato volba v nabídce Proces) dochází k postupné zobrazování jednotlivých položek slovníku, kdy se rychlost řídí posuvníkem v hlavním panelu. Interval je možné nastavit v rozmezí 0,5 s - 10 s. Uživatel v tomto režimu vyslovuje předkládaná slova, pokud je slovo rozpoznáno, je zvýrazněno zelenou barvou. To umožňuje vyzkoušení přesnosti rozpoznávače na sérii daných položek určitého slovníku.

Vedle systémového času je v tomto režimu zobrazován po řadě celkový čas prezentace, celkový počet slov v testu, zbývajících počet slov v testu a počet správně rozpoznaných slov v testu.

4.2 Grafické rozhraní

Hlavní okno programu je zobrazeno na obrázku 8. Okno aplikace obsahuje základní prvky běžné v aplikacích běžících pod MS Windows. Okno je rozděleno na menu, hlavní část, kde probíhají samotné testy a stavový pruh ve spodní části.



Obrázek 8: Okno aplikace

Menu obsahuje položky pro načtení konfiguračního souboru. Volba „Proces“ obsahuje položky, které zajišťují:

- Připojení k serveru / odpojení od serveru
- Začátek / konec nahrávání
- Volba režimu

Spodní pruh je vyhrazen pro informace o stavu serveru. Program naslouchá, když je v pravém dolním rohu zobrazena červená ikona REC.

Hlavní okno je rozděleno do série záložek, jejichž změnou se zároveň načítají slovníky očíslované na základě pořadí dané záložky (první záložka načítá soubor xxxx_1.vcb atd.) Počet záložek lze měnit a rozšiřovat tak sérii testů.

5 Testy

Testování probíhalo v deseti různých lokalitách za použití čtyř různých základních typů připojení k internetu. Celkově se testu zúčastnilo 43 osob (28 mužů, 15 žen) ve věkovém rozmezí 18 – 68 let. Zároveň byla otestováno chování rozpoznávače při mírně zvýšené hladině okolního hluku.

Ve všech testech byl použit stejný headset běžné kvality, zvukové karty integrované na základní desce každého daného PC. Internetové připojení zahrnovalo pevnou telefonní linku (ADSL a dial-up), mobilní připojení a různé typy širokopásmového připojení.

5.1 Sada testů

Byla připravena sada několika jednotlivých testů prostřednictvím testovací aplikace. Testy zahrnovaly čtení jednotlivých slov a krátkých frází, práci s pohyblivým oknem, prohlížení obrázků a spouštění externích aplikací. Celkem proběhlo šest testů a délka samotného testování nepřekročila 20 minut včetně úvodní instruktáže. Testy byly přizpůsobeny na základě skladby cílové skupiny, neobsahovaly tak například žádná cizí slova (vyjma vlastních názvů) a jednalo se celkově o jednoduché úkoly nevyžadující žádné hluboké studium jak samotné aplikace, tak i principů rozpoznávání řeči.

Jednalo se o tyto testy:

1) Čtení textu

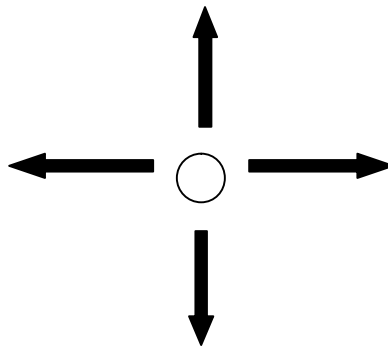
Slovník čítající 50 položek se skládal z názvů dnů v týdnu, měsíců, barev a dalších podstatných jmen. Uživatelé četli předlohu jak v normálním režimu, tak i v režimu prezentace.

2) Změna barvy obrázku

Byl zobrazen obrázek vozidla, povel bylo možné měnit jeho barvu (obrázek 11)

3) Pohyb obrazcem po obrazovce

Na povel se na obrazovce zobrazil obrazec, kterým bylo možno pohybovat ve čtyřech směrech (obrázek 9)



Obrázek 9: Pohyb objektem po obrazovce ve čtyřech směrech

4) Prohlížení fotografií

V okně externí aplikace byl simulován prohlížeč obrázků. Pomocí hlasových povelů bylo možné listovat fotografiemi vpřed a vzad.



Obrázek 10: Prohlížeč fotografií

Irak	Filipiny	Laos
Turecko	Keňa	Kapverdy
Rwanda	Hlavní město: Tunis	Slovensko
Malta	Litva	Bělorusko



Obrázek 11: Ukázka testů Zobrazení názvů hlavních měst a Změna barvy obrázku

Testování DSR

Test 1 | Test 2 | Test 3

Vyplnění formuláře

Osobní údaje:

Jméno ●

Příjmení

Věk

Roč.

Pohlaví ☐ Muž ☐ Žena

Bydliště:

Ulice

Číslo popisné

Město

Směrovací číslo

Rozeznávané slova

Obrázek 12: Vyplnění dotazníku

5) Zobrazení názvů hlavních měst

Zobrazení hlavního města na základě vyslovení jména státu (obrázek 11)

6) Vyplnění formuláře hlasem

Vyplnění formuláře s osobními údaji a údaji o bydlišti hlasem. Viz obrázek 9. Uživatel je postupně veden v uživatelském rozhraní od shora dolů, právě aktivní položka, která čeká na vyplnění je označena zeleným bodem. Když dojde ke kompletnímu vyplnění, je zobrazena souhrnná informace.

5.2 Výběr uživatelů

Uživatelé byli vybráni z řad mírně pokročilých v oblasti práce s počítačem. Každý mluvčí byl nejprve seznámen s účelem celé procedury a upozorněn na případné výrazy ve slovníku, které by mohli činit problémy při vyslovování. Dále byl poučen na možnost kdykoliv test přerušit a navázat později. Poté byl proveden testovací aplikační a byly mu zodpovězeny případné dotazy.

Testované osoby byly rozděleny podle věku do kategorií:

- 18 - 24 let
- 25 - 34 let
- 35 - 45 let
- nad 45 let

Pouhých 10% uživatelů mělo alespoň minimální zkušenosti s hlasovým ovládáním.

5.3 Dotazník

Dotazník je složen ze 12 základních položek (Obrázek 9)

Zvolena byla čtyřbodová hodnotící stupnice podle schématu:

1 – rozhodně ano, **2** - spíše ano, **3** - spíše ne, **4** - rozhodně ne

Po vyplnění dotazníku následoval krátký rozhovor s uživatelem, který mohl sdělit své názory jak na průběh testů, tak i své dojmy z celé procedury.

V kolonkách Test1 – Test4 byly na místě testu vepsány konkrétní položky vztahující se k daným testům.

Testování DSR rozpoznávače

Vyplňte prosím krátký dotazník, křížky označte svoji volbu.

Pohlaví Muž ☐ Žena ☐

Věk 18 - 24 ☐ 25 - 34 ☐ 35 - 45 ☐ 45 a více ☐

Ovládali jste někdy počítač hlasem ? Ano ☐ Ne ☐

Připojení k internetu:

Rozhodně NE				ê
Spíše NE			ê	
Spíše ANO		ê		
Rozhodně ANO	ê			

1. Reakce programu jsou okamžité ☐ ☐ ☐ ☐

2. Ovládání vyžaduje velké soustředění ☐ ☐ ☐ ☐

3. Ovládání programu tímto způsobem mi vyhovuje: ☐ - ☐ - ☐ - ☐

4. Test1: ☐ ☐ ☐ ☐

5. Test2: ☐ ☐ ☐ ☐

6. Test3: ☐ ☐ ☐ ☐

7. Test4: ☐ ☐ ☐ ☐

8. Program občas reaguje zcela nečekaně. ☐ ☐ ☐ ☐

9. Při zvýšené hladině hluku je ovládání nepoužitelné ☐ ☐ ☐ ☐

10. Nevím, jak vyslovovat dané fráze ☐ ☐ ☐ ☐

11. Program často vůbec nereaguje ☐ ☐ ☐ ☐

12. Chtěl(a) bych takto ovládat i jiné programy. ☐ ☐ ☐ ☐

Obrázek 13: Dotazník

Přiřazení testů 1 – 4 v dotazníku bylo následující:

Test1 - Změna barvy obrázku

Test2 – Pohyb objektem po obrazovce

Test3 – Vyplnění formuláře

Test4 – Prohlížení obrázků

Otázka v dotazníku byla pro tyto testy společná: Ovládání programu tímto způsobem mi vyhovuje.

6 Výsledky

S pomocí dané aplikace použité při testech již bylo možné vyvodit závěry, které jsou zaneseny ve slovním hodnocení a vybrané v grafech. Byly provedeny i pohovory po ukončení používání testovací aplikace.

Uživatelé se většinou rychle sžili s ovládáním. Přesto se samozřejmě vyskytovaly problémy, zvláště u velkých slovníků, podobně znějících slov a zvýšené hladině okolního hluku. To mělo za následek například samovolné spouštění aplikací.

Vybrané výsledky následují v grafech. Zbytek je podrobněji rozebrán dále v položkách jednotlivých testů.

6.1 Čtení textu

Tento test sloužil jako úvodní společně s testem následujícím. Hlavním cílem bylo uživateli předvést funkci aplikace a nechat ho vyzkoušet si práci s rozpoznávačem, aniž by program vyvolával nechtěné reakce. V nejnepříznivějším případě došlo pouze k chybnému zobrazení rozpoznaného slova.

6.2 Změna barvy obrázku

Slovník čítající jen několik položek (názvy barev) se ukázal jako vhodný pro takovouto aplikaci. Změna barev většině uživatelů nečinila potíže a ovládání tohoto testu jako vyhovující označilo 90% uživatelů.

6.3 Pohyb objektem po obrazovce

I tento test nečinil téměř žádné potíže. Velmi malý slovník (čtyři směry) znatelně snižoval možnost špatné interpretace promluvy a tak jako vyhovující ho označili všichni uživatelé.

6.4 Zobrazení názvů hlavních měst

Jednalo se o zobrazení hlavního města daného státu ve stylu televizní soutěže. Vzhledem k celkem odlišným vlastním názvům států vykazoval rozpoznávač minimální chyby. Za plně vyhovující tento systém v konkrétním případě označilo 93% uživatelů.

6.5 Prohlížení obrázků

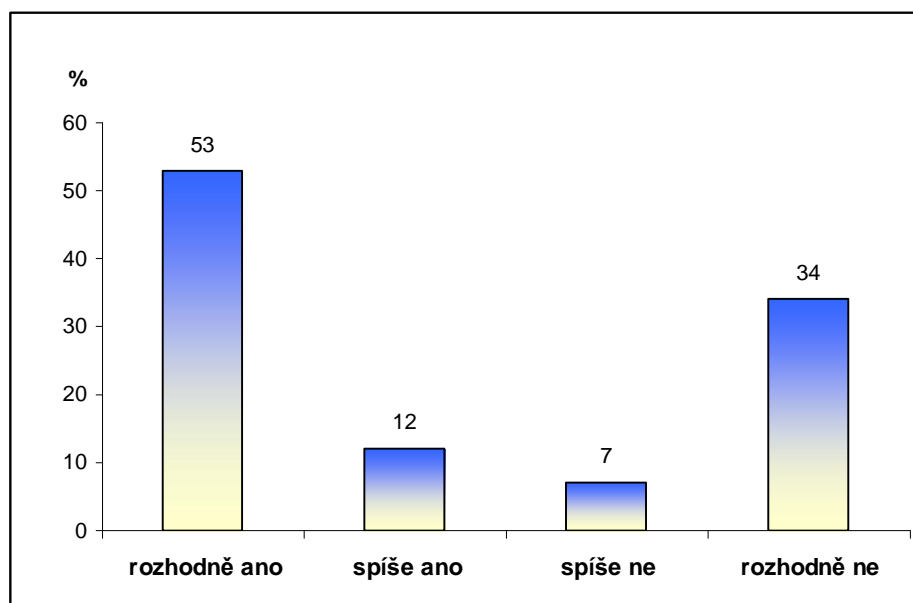
Tato aplikace hlasového rozpoznávače se ukázala jako vhodná. Posouvání obrázky vpřed a vzad pomocí hlasu označili vyjma jednoho uživatele všichni za vyhovující.

6.6 Vyplnění formuláře

Tento test představoval nejnáročnější zkoušku pro uživatele. Vzhledem k většímu slovníku a podobnosti např. jmen a měst docházelo často k záměně. Tento způsob se rozhodně nestal oblíbeným a za vyhovující ho označilo pouze 23% uživatelů.

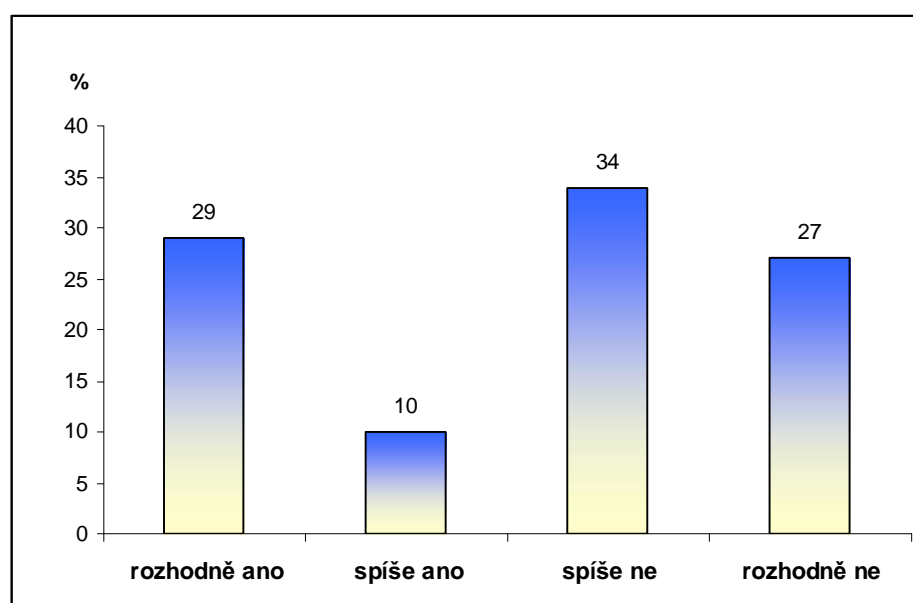
Následují graficky zobrazené vybrané výsledky z dotazníku.

6.1 Reakce programu



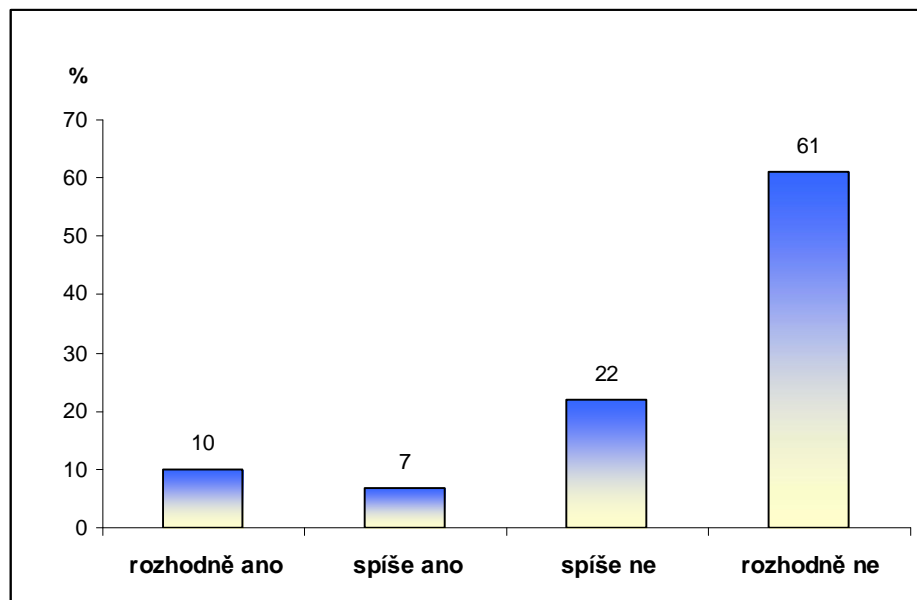
Obrázek 14: Názor uživatelů: reakce programu jsou okamžité

6.2 Program reaguje zcela nečekaně



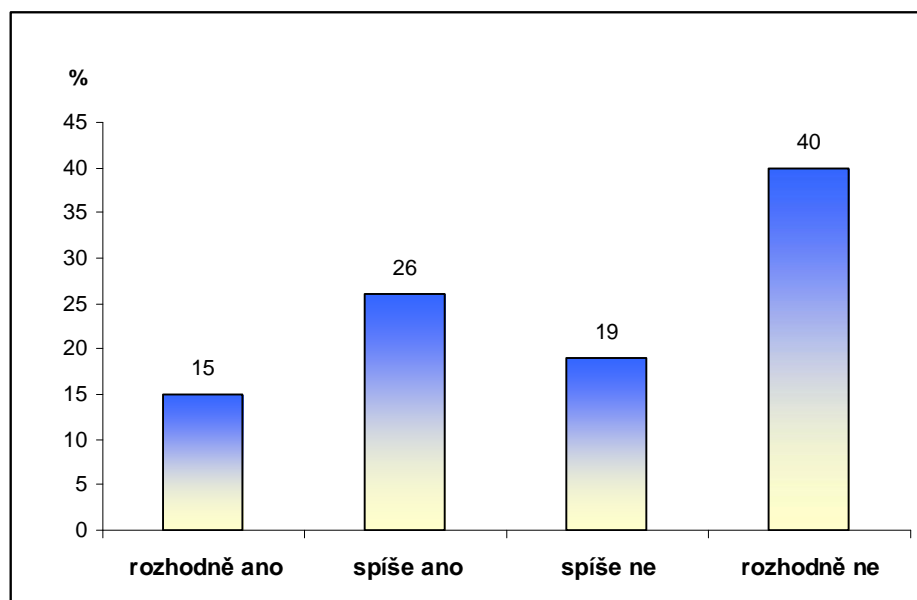
Obrázek 15: Názor uživatelů: program reaguje zcela nečekaně

6.3 Ovládání vyžaduje velké soustředění



Obrázek 16: Názor uživatelů: ovládání vyžaduje velké soustředění

6.4 Nevím, jak vyslovovat dané fráze



Obrázek 17: Názor uživatelů: Nevím, jak vyslovovat dané fráze

Závěr

S pomocí testovací aplikace byla otestována skupina 43 uživatelů a na sadě testů byla pozorována jejich práce s rozhraním pomocí hlasového vstupu.

Aplikace běžící s daným distribuovaným rozpoznávačem izolovaných slov jsou použitelné zejména v případech, kdy není kladen důraz na co nejrychlejší odezvu systému. Vzhledem k nutnosti internetového připojení nemůže být tato odezva vždy zaručena, na druhou stranu přístup ke kvalitní internetové lince je dnes již většinou samozřejmostí. Větším problémem je okolní ruch, který činil potíže na otevřených prostranstvích a v místnostech, kde byl zapnutý např. televizor. To také mnohé uživatele zprvu mátló neboť rozpoznávač někdy zaznamenal i neřečové události, což mělo za následek vykonání náhodného povelu. To bylo podle uživatelů nejvíce matoucí.

Podle předpokladů byly do značné míry výsledky ovlivněny velikostí konkrétního slovníku v daném testu, dále použitým typem internetového připojení a také celkovým přístupem konkrétního uživatele. Někteří totiž stále ještě této technologii „nevěří“, a snaží se přehnaně artikulovat v domněnku, že tím zvýší úspěšnost rozpoznání systému.

Do budoucna by bylo vhodné aplikaci rozšířit a postavit na ní reálný systém, který bude plnit skutečnou funkci vhodnou pro studenty (např. objednávání obědů v menze, obsluha kopírovacího systému apod.) Na tomto systému by pak bylo možné aplikovat podobnou sadu testů a v rámci delšího testovacího cyklu (nebo cyklů) vyvodit další závěry a kritéria pro návrh aplikací tohoto typu.

Přestože jsou aplikace na bázi hlasových rozhraní stále více rozšířené, většina uživatelů, kteří se testů zúčastnila nikdy tuto možnost ovládání nevyužila. Do jisté míry je pak test ovlivněn cílovou skupinou, kdy dojdeme k odlišným výsledkům při výběru odlišného vzorku populace. Zatímco začátečníci mohou přehlédnout chyby nebo přehnaně reagovat na nepodstatné věci, experti v dané oblasti dokážou ocenit kladné stránky i odhalit skryté chyby i zjevné nedostatky daného systému.

Literatura

- [1] Holada, M.: Internet Speech Recognition Server. In *Proc. of 7th World Multiconference on Systemics, Cybernetics and Informatics-SCI 2003*. Orlando-USA, July 2003. Volume IV. pp. 388-391. ISBN 980-6560-01-9
- [2] Miroslav Holada, Martin Pelc: Optimizing Distributed Speech Recognition System for Multi-User Real Time Usage, In: *Speech and Computer International Conference - Specom 2007*, October, 2007, Moscow, Russia, pp. 455-459, ISBN 6-7452-0110-X
- [3] Nouza, Jan: Počítačové zpracování řeči, Liberec : Technická univerzita v Liberci, 2001, ISBN 80-7083-551-6
- [4] Miroslav Holada, Martin Pelc (2008). The Robot Voice-control System with Interactive Learning, *New Developments in Robotics Automation and Control*, Aleksandar Lazinica (Ed.), ISBN: 978-953-7619-20-6, InTech, Austria, Available from:
<http://sciyo.com/articles/show/title/the_robot_voice-control_system_with_interactive_learning>
- [5] Psutka, Josef: Komunikace s počítačem mluvenou řečí, Praha : Academia, 1995, ISBN: 80-200-0203-0
- [6] DUNDIS: Internetový řečový rozpoznávač [online] [cit. cit. 2010-4-20] Dostupný z: <<https://www.ite.tul.cz/speechlab/index.php/stari-projekty/dundis.html>>
- [7] Hone, K. S. and Golightly, D. (1998) Interfaces for speech recognition systems: the impact of vocabulary constraints and syntax on performance. *Proceedings of the 5th International Conference on Spoken Language Processing*, Vol. 4, pp. 1199-1202.
- [8] Schulz, Stefan / Donker, Hilko (2006): "An user-centered development of an intuitive dialog control for speech-controlled music selection in cars", In *INTERSPEECH-2006*, paper 1855-Mon1A3O.4.
- [9] Markku Turunen. *Speech Application Design and Development*. On-line book, 87 str., [online] February 2004 [cit. 2010-2-10].
Dostupný z : <<http://www.cs.uta.fi/hci/spi/reports/SADD.pdf>>
- [10] NIELSEN, Jakob. Quantitative studies: How many users to test. [online]. 2000 [cit. 2010-5-18].

Dostupný z WWW: <http://www.useit.com/alertbox/quantitative_testing.html>.

[11] NIELSEN, Jakob. Summary of Usability Inspection Methods [online]. 1994 [cit. 2010-5-18]. Dostupný z WWW:

<http://www.useit.com/papers/heuristic/inspection_summary.html>.

[12] NIELSEN, Jakob. Technology Transfer of Heuristic Evaluation and Usability Inspection

[online]. 1995 [cit. 2010-5-3]. Dostupný z WWW:

<http://www.useit.com/papers/heuristic/learning_inspection.html>.

[13] Larsen, Lars Bo (2003): "Assessment of spoken dialogue system usability - what are we really measuring?", In *EUROSPEECH-2003*, 1945-1948.

[14] Patrick W. Jordan, Bruce Thomas, Bernard A. Weerdmeester, Ian McClelland: *Usability Evaluation in Industry*. Taylor & Francis Ltd. 1996. ISBN:0748404600

[15] What is SUMI?[online].[cit.17.4.2010].

Dostupný z <<http://sumi.ucc.ie/whatis.html>>.

[16] VÍŠEK, O. Použitelnost aplikačního rozhraní GeoWebů Pardubice: Univerzita Pardubice, 2007. 74 s. Diplomová práce.

[17] Turunen, Markku / Hakulinen, Jaakko / Melto, Alekski / Heimonen, Tomi / Laivo, Tuuli / Hella, Juho (2009): "SUXES - User Experience Evaluation Method for Spoken and Multimodal Interaction", In *INTERSPEECH-2009*, 2567-2570.

[18] A. WATSON and MA SASSE, Measuring perceived quality of speech and video in multimedia conferencing applications. *Proceedings of ACM multimedia 98*, ACM (1998) p. 55–60 – měření kvality porozumění zvuku

[19] Shneiderman's "Eight Golden Rules of Interface Design" [online]

[cit. 2010-3-8]. Dostupný z

<<http://faculty.washington.edu/jtenenbg/courses/360/f04/sessions/schneidermanGoldenRules.html>>

Přílohy v elektronické formě

DVD-R

- - text práce ve formátu .doc a .pdf
- -zdrojové soubory aplikací
- -slovníky
- sady obrázků pro vybrané testy